VA Office of Information and Technology Enterprise Architecture Management

Systems Integration and Development Service

Configuration Identification Procedure CM310

Version 1.0
May 8, 2006

VA Office of Information and Technology
Enterprise Architecture Management
Systems Integration and Development Service

Letter of Promulgation

As the Director of the Systems Integration and Development Service (SIDS) within the Office of Enterprise Architecture Management, Department of Veterans Affairs, Office of Information and Technology (OI&T), I do hereby formally promulgate and approve this "Configuration Identification Procedure" and direct its use across the SIDS. "Identification" in this context refers to the selection, definition, and application of unique identifiers to product items, components and elements to be placed under formal configuration control. This procedure establishes, defines, and describes the rules and systematic approach for configuration item identification within the SIDS, to SIDS assets, and to assigned SIDS programs and projects, and ultimately helps ensure the integrity of SIDS and the OI&T products.

(Signature obtained and on file)
Frances G. Parker, Director (Acting)
Systems Integration and Development Service
Office of Enterprise Architecture Management

May 8, 2006
(Date)

Record of Changes

CCP #	CCP Date	Description of Change (or title)	Date Entered	Entered by: (initials)
N/A	5/8/06	Initial Issue	5/8/06	bgl

Table of Contents

1. INTRODUCTION

1.1. PURPOSE

This procedure establishes the process, steps, and guidance for identifying configuration items for formal configuration control.

1.2. SCOPE

This instruction applies to all items of all tasks and projects assigned to the Systems Integration and Development Service (SIDS) organization. "Identification" in the context of configuration management refers to the selection, definition, and application of unique identifiers to product items, components and elements to be placed under formal configuration control. Approval of a written Request for Waiver (RFW) to the Director, SIDS, is required for non-compliance with any section or subsection of this document. An RFW may be submitted by email until such time as an official RFW procedure and template are adopted by SIDS.

1.3. AUTHORITY

The Director, SIDS, is the issuing authority for this document and only the Director, SIDS, or higher authority may authorize it to be altered, superseded, or cancelled. Any changes or modifications to this document must be submitted for approval in accordance with SIDS Configuration Change Management Procedures.

Any conflict between this document and higher authority will be resolved in favor of the higher authority. Anyone observing such a conflict is requested to bring it to the immediate attention of the Director, SIDS (or delegated SIDS authority).

1.4. CHANGE AND CANCELLATION

This is an original document and does not supersede a previous version or any other document.

1.5. REFERENCES

The following references were used in developing this procedure:

1) Carnegie-Mellon University/Software Engineering Institute, Capability Maturity Model Integration for Systems Engineering/Software/Integrated Products and Processes Development/Supplier Sources, March 2002.
2) Clinger-Cohen Act of 1996 (Public Law 104-106).
3) Department of Veterans Affairs Directive 6000.
4) Electronic Industries Alliance /Government Electronic and Information Technology Association Industries Alliance (EIA/GEIA) Standard-649-A, National Consensus Standard for Configuration Management, April 2004.

1.6. DEFINITIONS AND ABBREVIATIONS

1.6.1. Definitions

Refer to the SIDS Configuration Management (CM) Plan.

1.6.2. Abbreviations

The following abbreviations are used in this procedure. This table does not include abbreviations used as examples or possible data entries in identifiers.

CCMB	-	Configuration Change Management Board
CI	-	Configuration Item
CM	-	Configuration Management
ICD		Interface Control Document
RFW	-	Request for Waiver (or Deviation)
SIDS	-	Systems Integration and Development Service

1.7. ASSUMPTIONS

This procedure assumes that users have a working knowledge of:

a. Product disassembly into functional components and parts,
a. Criticality and safety risk determination, and
b. Product description attributes.

2. PROCESS DIAGRAM

Figure-1 illustrates the sequence of steps in the identification process.

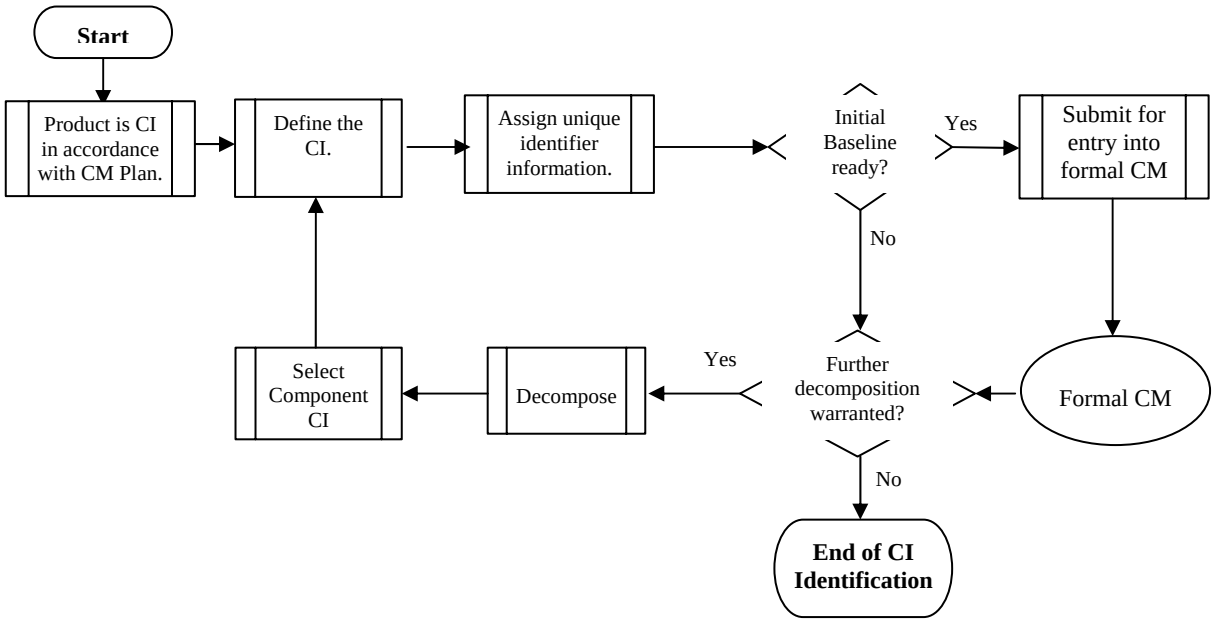

Figure 1 – Configuration Identification Process Flow Diagram

3. PROCEDURE

This section presents the process steps and guidance for configuration item (CI) identification.

3.1. DISCUSSION

A CI is any product or part of a product designated (selected, defined, and marked or labeled with a unique identifier) for formal configuration change management and control. CI Identification is the process for making those designations. The work products and artifacts to be placed under formal CM include products to be delivered, designated internal items, acquired products, tools, other items used in creating and describing these items, and management items for the project organization.

CI can be decomposed into configuration components, units, and elements. While these descriptive terms illustrate a hierarchical relationship of the parts, each component, unit or element can be designated as a CI and decomposed further. Therefore, in this procedure and related guidance documents, the term "configuration item" (or "CI") may be interpreted as "configuration component," "configuration unit," or "configuration element."

3.2. CI IDENTIFICATION STEPS

A checklist for the steps of configuration identification is contained in Attachment 1, CI Identification Checklist. The following paragraphs contain guidance for addressing the steps. The checklist addresses the process flow for large, complex products and projects as illustrated in Section 2, Figure 1.

3.2.1. Establish the Initial CI

In accordance with the SIDS Configuration Management Plan, each product for which SIDS is responsible is a CI. Therefore, the product is pre-selected and should immediately be defined and assigned a unique identifier in accordance with Sections 3.2.4 and 3.2.5 below. It then must be determined if the product CI will contain or be comprised of other lower-level CI by returning to and applying Section 3.2.2.

3.2.2. Decomposition

Decomposition of a CI refers to separating a major product or artifact into multiple pieces to yield sub-products. It is often accomplished during a design stage of a System Development Life Cycle. The results of decomposing a product will be such individual items as plans, procedures, process descriptions, requirements, design data, drawings, product specifications, database designs, code, compilers, product data files, interfaces between two or more individual items, and product technical publications. Effective management of these individual items will result in effective management of the product.

Using a software product as an example, it can be decomposed into a second tier of 1) project management items and 2) product items. Project management can be decomposed into a third-tier of documents for such functions as Project Plan, Requirements Management Documents, Risk Management Documents, etc. A fourth tier under Risk Management

Documents might be the Risk Management Plan, various Risk Management procedures, and a Risk Mitigation Plan Template. Often, these are already established as management CI by the organization and require no action by the project management team.

The other second-tier group, product items, can be decomposed into a third tier of Requirements, Manuals, Designs, Source Code Files, etc., and "Manuals" into a fourth tier of User's Manual, Installation Guide, Tutorial Manual, etc. The first sub-product for the project team will often be a Requirements Traceability Matrix developed from a functional and business requirements document, which is owned by the business line or "customer." All other CI within the product will derive from the Requirements Traceability Matrix.

Refer to SIDS procedure CM311, SIDS Product Decomposition Guide.

3.2.3. Selection

After decomposition, it may be determined that some of the sub-products or components require formal configuration control while others should be simply managed. Selection is the use of defined criteria to make the decision or determination that an item will be a CI, i.e., managed under formal configuration control. Interfaces must always be selected as CI.

If a resulting smaller part or item is still considered to be too large or too complex for effective management, it should be defined, have an identifier affixed, and then decomposed further. Once the criteria has been applied and an item has been selected as a CI, it must be reviewed with regard to its manageability to determine if it can be effectively managed at this level of detail or if it would be better to decompose it into yet smaller parts.

Refer to SIDS procedure CM312, SIDS Configuration Item Selection Guide.

3.2.4. Definition

An item that has been selected as a CI must be defined in terms of its descriptive and management attributes. Refer to SIDS procedure CM313, SIDS Configuration Item Definition Guide. Definition of the CI proactively:

- Creates responsibility for and authority over each CI, whether the CI is at the element level or is a grouping of elements,
- Provides a mapping of the hierarchy of the parts of the top-level CI,
- Facilitates proper placement of the file(s) in the CM library, and
- Gives the interested stakeholders a quick summary of the nature of the CI.

The attributes that comprise the definition of a CI include such information as:

- Filename,
- Parent item,
- Parent project/product group,
- Purpose (management, script, dml, mml, etc.),
- File type (drawing (Visio, Clipart, etc.), design (Win draw, Corel draw, ERWin, etc.), document, source code),

- Programming language and operating system (for software code files),
- Database management system and operating system (for databases),
- Interface (y/n) and interdependent items,
- Non-interface items related such that a change in one mandates a change in the others,
- Author,
- Change authorizer (role),
- Change incorporation responsibility (role or group).

It may be necessary to record some of this information on the properties sheet of electronic files. Interface control documents (ICD), by design, will contain additional definition attributes for an interface. (Refer to SIDS procedure CM315, Interface Control Procedure.)

3.2.5. Identifier Application

An item that has been selected as a CI must have a unique means of identification to preclude confusing it with any other CI with one or more similar attributes. The establishment of configuration control implies that a CI will undergo changes. Therefore, the unique identifier also must include the means to differentiate between revisions of the item.

When the CI is a product (not a component, unit or element), the basic rules for applying an identifier are as follows:

1. Each product is assigned a unique identifier by its developer (with the aid of the assigned CM specialist).
2. Existing product, used without change as components of another product, retain the original identifiers in their new use.
3. When a product is changed, its identifier is updated to reflect the new configuration, when:
    - The new or updated product is no longer interchangeable functionally or physically with the product it replaces;
    - The new product requires new or revised testing, maintenance, repair training, operating procedures, equipment, or software;
    - The product is altered, selected or is a source (vendor) controlled item; or
    - The updated product has different restrictions (e.g. application, safety, etc.).

When multiple copies of a configuration component, unit or element are required (that is, can be correlated to a process, date, event, or test), the CI should be assigned a “batch” or lot identifier to facilitate traceability to the correct group of copies in the event of latent defect discovery. This provision for batch identification is more normally experienced in a hardware environment. (Examples include such items as computer hard drives, special nuts and bolts, and computer or non-computer cabling).

Refer to SIDS procedure CM314, SIDS Unique Identifiers Instruction for the formulation of unique identifiers.

ATTACHMENT 1 – CI IDENTIFICATION CHECKLIST

It may be necessary to return to this procedure and checklist during the evolution of a product throughout the development life cycle. Such a return will probably commence with Step 6.

_____ 1. Project Manager, receive the "product" by assignment (contract, direction, assigned functional requirements, etc.). By definition, this is selected as a CI.

_____ 2. Project Manager, with other project personnel, define the CI. (Refer to SIDS CM313, CI Definition Guide.)

_____ 3. Project Manager, or delegated authority, and the assigned CM Facilitator develop and assign the unique identifier to the product. (Refer to SIDS CM314, Unique Identifiers Instruction.)

_____ 4. Project Manager (or delegated authority), determine if the initial baseline is ready for submittal to the CM Facilitator for formal configuration control.

_____ a. If the initial baseline is ready, submit to the CM Facilitator for initiation of configuration control and continue with Step 5. (Refer to SIDS procedure CM422, Baseline Submittal Procedure.)

_____ b. If the initial baseline is not ready, continue to work on the baseline AND continue with Step 5.

_____ 5. Project Manager, with the CM Facilitator and other team members, determine if further decomposition of the CI is necessary for better control.

_____ a. If no further decomposition is required, go to Step 9.

_____ b. If further decomposition is appropriate, go to Step 6.

_____ 6. Project Manager, project team and CM Facilitator, decompose the CI into components or elements. (Refer to SIDS CM311, CI Decomposition Guide.)

_____ 7. Project Manager and project team, with CM Facilitator, select the items for formal configuration control. (Refer to SIDS CM312, Configuration Item Selection Guide.)

_____ 8. Return to Step 2.

_____ 9. End of CI Identification procedure. Proceed with configuration control in accordance with configuration change management procedures.